TP 1 : Commandes de base sous Unix

Les commandes de base

- La commande essentielle : man nom_commande (aide en ligne)
- Les commandes primaires :
  - **ls nom_repertoire** : liste les fichiers du répertoire spécifié
  - **cd nom_repertoire** : change le répertoire courant pour le répertoire spécifié
  - **pwd** : affiche le chemin du répertoire courant
  - **mkdir nom_repertoire** : crée un répertoire
  - **rmdir nom_repertoire** : supprime un répertoire
  - **cp nom_fichier autre_nom_fichier** : crée un double du fichier avec un nouveau nom
  - **mv nom_fichier autre_nom_fichier** : renomme le fichier
  - **mv nom_fichier nom_répertoire** : déplace le fichier dans le répertoire indiqué
  - **rm nom_fichier** : supprime le fichier
- Les commandes de visualisation :
  - cat nom_fichier : visualisation du contenu d'un fichier
  - more nom_fichier : filtre de pagination

Toutes ces commandes admettent de nombreuses options (précisant des utilisations différentes).

Exercice 1

1. Utilisez l'aide en ligne et expérimentez chacune des commandes ci-dessus
2. Visualisez le contenu actuel de votre compte grâce à la commande **ls -l**. En vous aidant de l'aide en ligne essayez de comprendre chacune des informations fournies
3. Créez un répertoire **SE** à la racine de votre compte
4. Déplacez-vous dans ce répertoire
5. Créez un fichier vide grâce à la commande : **touch nom_fichier1**
6. Créez un fichier non-vide en utilisant : **cat > nom_fichier2** puis entrez un texte sur plusieurs lignes et terminez en tapant simultanément les touches **Ctrl-D**
7. Visualisez le contenu de ce dernier fichier avec les commandes proposées ci-dessus
8. Créez un sous-répertoire **TP1**
9. Visualisez le contenu de votre répertoire courant grâce à la commande **ls -l**
10. Visualisez maintenant le contenu de ce même répertoire grâce à la commande **ls -al**
11. Effacez le fichier vide.
12. Renommez l'autre fichier en **toto**
13. Créez une copie de **toto** dans le répertoire **/home/user**
14. Vérifiez que la copie est bien dans **/home/user**
15. Déplacez toto dans **TP1**
16. Essayez d'effacer le répertoire **TP1**. Que se passe-t-il ?
17. Créez une archive du répertoire **TP1** dans le répertoire **/home/user** (Utilisez la commande **tar**)

Typeset by Hicham LAANAYA with LaTeX September 28, 2018

Scanned with CamScanner

Exercice 2

1. Créer le répertoire **TP2** dans votre répertoire **SE**

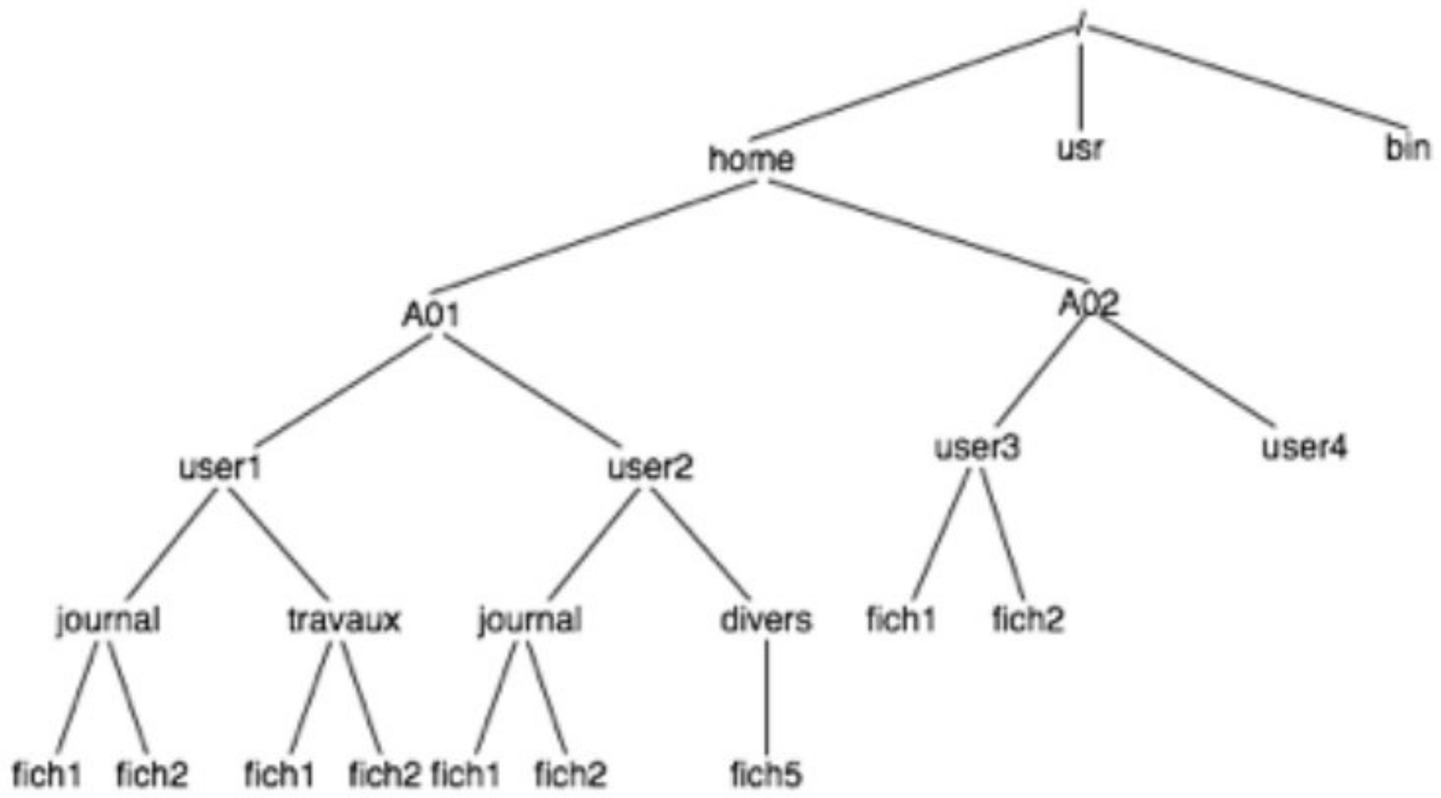

Figure 1 : Exemple d'une arborescence

2. Créer la sous-hiérarchie présentée par la figure 1 depuis le répertoire **`home`** en la faisant démarrer du repertoire **`TP2`**

3. Vérifier votre hiérarchie en utilisant la commande **`ls -R`**

4. Effectuer les commandes suivantes en supposant que l'utilisateur se trouve actuellement dans le sous-répertoire **`journal`** du répertoire **`user1`**

   (a) Lister le contenu du répertoire courant

   (b) Lister le contenu du répertoire **`travaux`**

   (c) Afficher le chemin du répertoire courant

   (d) Créer un répertoire **`lundi`**

   (e) Créer un fichier vide **`cr.txt`**

   (f) Copier le fichier **`fich1`** dans travaux en le nommant **`fich3`**

   (g) Copier le fichier **`fich5`** du répertoire **`user2/divers`** dans le répertoire courant

   (h) Renommer ce fichier en **`fich_user2`**

   (i) Déplacer le fichier **`fich2`** du répertoire courant dans le répertoire **`user1`**

   (j) Déplacer le fichier **`fich2`** du répertoire **`travaux`** dans le répertoire **`lundi`** en le renommant **`urgent`**

   (k) Dessiner la nouvelle arborescence obtenue après ces opérations

   (l) Vérifiez que vous avez bien répondu à la question précédente grâce à la commande **`ls -R`**

Scanned with CamScanner

Université Mohammed V — Rabat
Faculté des Sciences Rabat
Département d'informatique

2017—2018
Systèmes d'Exploitation 3
SMI — S3

TP 2 : Droits d'accès et liens

1 Droits d'accès des fichiers

Étant donné qu'Unix est un système d'exploitation multi-utilisateurs, on distingue trois catégories d'utilisateurs :

- **u** : propriétaire (user : celui qui a crée le fichier).
- **g** : le groupe (les autres utilisateurs appartenant au même groupe du propriétaire).
- **o** : les autres (others).

Ainsi, chaque fichier possède trois types de droits :

- **r** : lecture (read) ;
- **w** : écriture (write) ;
- **x** : exécution (execute) ;
- — : aucun droit.

Au niveau répertoire, ces droits signifient :

- Droit de lister les fichiers présents dans ce répertoire (**ls**) ;
- Droit de créer ou de détruire un fichier se trouvant dans ce répertoire ;
- Droit d'accéder à ce répertoire (**cd**)

Exercice 1

1. Déterminez votre ou vos groupes d'appartenance (**id**).
2. Utilisez **ls -l** depuis votre répertoire personnel, et repérez les symboles décrivant les droits, le propriétaire et le groupe d'appartenance du fichier.
3. Donnez les équivalences symboliques des droits d'accès suivants : `744, 633, 755, 111, 722` et `700`

Exercice 2

1. Créez un répertoire **test** et un fichier **essai** dans ce répertoire, et écrivez-y la phrase de votre choix.
2. Notez à l'aide de `ls -l` les permissions actuelles du répertoire **test** et du fichier **essai**.
3. En utilisant la commande chmod, retirez-vous le droit en lecture et en écriture sur le fichier **essai**. Vérifiez l'effet obtenu en essayant d'afficher le contenu du fichier sur la fenêtre du terminal, puis de remplacer ce contenu par une phrase différente.
4. Rétablissez le droit en écriture puis remplacez le contenu du fichier **essai** par le texte **echo "Ceci est un essai"**. Ajoutez-vous le droit en exécution, et exécutez le fichier **essai** en tapant **./essai** dans le terminal (depuis le répertoire qui le contient). Quel est le problème ?
5. Rétablissez enfin le droit en lecture et tentez à nouveau d'exécuter le fichier.
6. Placez-vous dans le répertoire **test**, et retirez-vous le droit en lecture pour ce répertoire. Listez le contenu du répertoire, puis affichez le contenu du fichier **essai**. Qu'en déduisez-vous? Rétablissez le droit en lecture sur **test**.

Typeset by Hicham LAANAYA with XeLaTeX October 2, 2017

Scanned with CamScanner

7. Créez un fichier nommé nouveau dans **test**. Retirez à ce fichier et au répertoire **test** le droit en écriture. Tentez de modifier le fichier nouveau. Rétablissez ensuite le droit en écriture au répertoire **test**. Tentez de modifier le fichier nouveau, puis de le supprimer. Que constatez-vous ?

8. Depuis votre répertoire personnel, retirez enfin le droit en exécution du répertoire **test**. Tentez de créer, supprimer, ou modifier un fichier dans le répertoire **test**, de vous y déplacer, d'en faire la liste, etc. Qu'en déduisez-vous quant au sens du droit en exécution pour les répertoires ?

Exercice 3 — umask

1. Donnez la commande pour connaître `umask`.
2. Donnez la commande pour changer umask à `044`.
3. Définissez un umask équilibré qui vous autorise un accès complet et autorise un accès en lecture aux membres de votre groupe. Testez sur un nouveau fichier et un nouveau répertoire.
4. Redémarrez votre ordinateur et vérifiez de nouveau la valeur de umask

Exercice 3 — Liens physiques et liens symboliques

1. Faites deux copies du fichier **/etc/passwd** l'une appelée **passwd1** et l'autre appelée **passwd2**.
2. Comparez leurs numéros d'inœuds à l'aide de la commande `ls -li`.
3. Créez un lien physique à l'aide de la commande `ln` pour le fichier **passwd1** appelé **passwdph**.
4. Comparez leurs numéros inœuds.
5. Modifiez le contenu du fichier **passwd1**. Que remarquez-vous à propos du fichier **passwdph** ?
6. Créez un lien symbolique pour le fichier **passwd2** appelé **passwdsy** à l'aide de la commande `ln -s`.
7. Comparez leurs numéros d'inœus.
8. Modifiez le contenu du fichier **passwd2**. Que remarquez-vous à propos du fichier **passwdsy** ?
9. Supprimez le fichier **passwd2**. Que se passe-t-il pour le fichier **passwdsy** ?
10. Editez de nouveau le fichier nommé **passwd2**. Que remarquez- vous ?

Scanned with CamScanner

Université Mohammed V—Agdal
Faculté des Sciences Rabat
Département d'Informatique

Année Universitaire 2014—2015
Systèmes d'exploitation 3
Filière : SMI - S3

TD 2 : Programmation Shell

Exercice 1

Ecrire un script Shell qui va calculer la somme de deux nombres, qui sont fournis en arguments de la commande, et si ces deux nombres ne sont pas passés en arguments, afficher une erreur et montrer l'utilisation de la commande.

Exercice 2

Ecrire un script pour connaitre le plus grand nombre de trois nombres donnés en paramètres de la commande. Afficher un message d'erreur si le nombre d'arguments n'est pas suffisant.

Exercice 3

Ecrire un script, utilisant une boucle `while`, qui va afficher
n
$n-1$
...
1

Exercice 4

Ecrire un script Shell permettant d'itérer une lecture au clavier jusqu'à obtention de la chaine `oui` *ou de la chaîne* `non`

Exercice 5 : utilisation de la commande `expr`

Dans le langage Bourne-Shell, il n'y a pas, à proprement parler, de variables numériques. Les manipulations arithmétiques sont réalisées au travers de la commande `expr` *qui interprète certains de ses paramètres de position comme les représentations ASCII de nombres entiers, les arguments incorrects provoquant une erreur de la commande* `expr` *matérialisée par un code de retour égale à* 2

1. *Par quel moyen simple est-il possible de tester qu'une variable Shell est une valeur interprétable numériquement?*
2. *Écrire un script, en utilisant* `case`, *qui effectue les opérations mathématiques :*
   - *L'addition* +,
   - *La soustraction* −,
   - *La multiplication*x,
   - *La division* /.

   Le nom de script doit être `script5` *qui fonctionne comme suit* `./script5 20 / 3`. *Vérifier que le nombre d'arguments est correct et que le 1-er et le 2-ème arguments sont interprétables numériquement.*

Exercice 6 : utilisation de la commande `exit`

Ecrire un script Shell appelable soit sans arguments, soit avec trois arguments et telle que, appelée sans arguments, il réalise la lecture au clavier de trois chaînes de caractères. Disposant alors dans tous les cas de trois chaînes, il indique si les trois chaînes sont identiques, si deux de ces chaînes sont identiques ou si elles sont toutes différentes par un message sur la sortie-erreur-standard (utiliser la commande `exit`*). Le code de retour de la commande sera égal à* 0 (`exit 0`) *si les trois chaînes sont égales,* 1 (`exit 1`), 2 (`exit 2`) ou 3 (`exit 3`) *selon que la chaîne en i-ème position est différente des deux autres (celles-ci étant identiques),* 4 *si les trois chaînes sont différentes et* 5 *si le nombre de paramètres d'appel est incorrect.*

Typeset by Hicham LAANAYA with XeLaTeX December 2, 2014

exosup.com
page facebook
Scanned with CamScanner